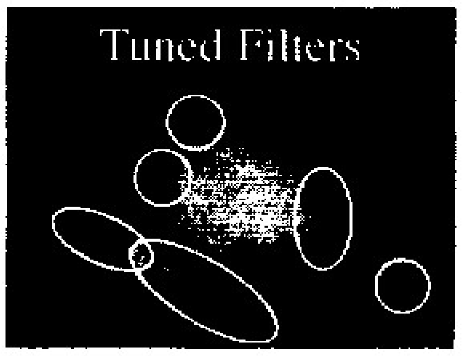
PROGRESS ON VISION THROUGH LEARNING AT GEORGE MASON UNIVERSITY

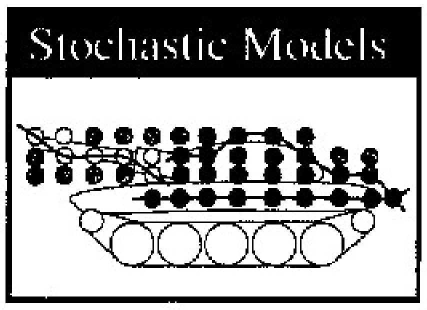
by

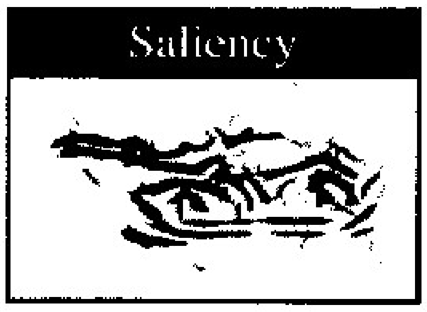
J. W. Bala
R. S. Michalski
P. W. Pachowicz

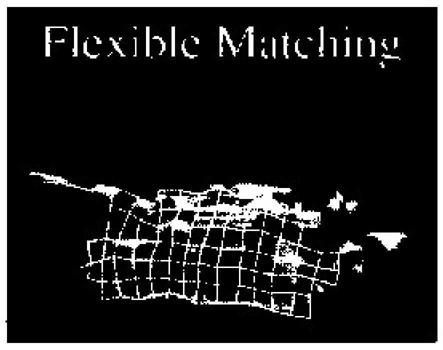
Proceedings of the ARPA Image Understanding Workshop, November 13-16, 1994.

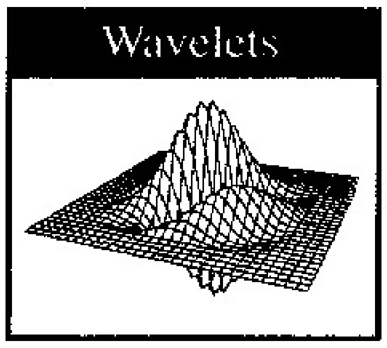
1994
Image Understanding Workshop







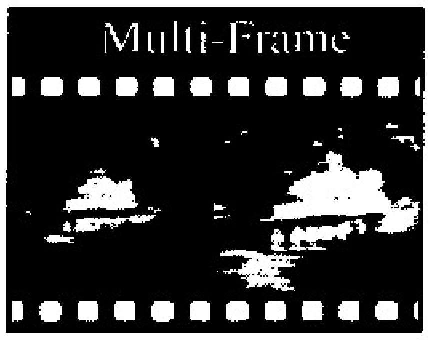


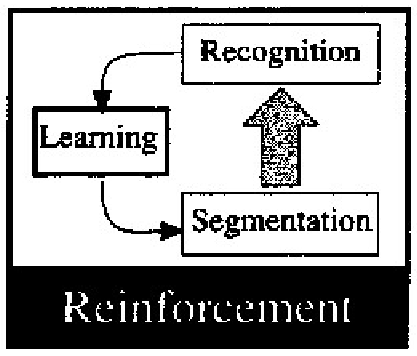


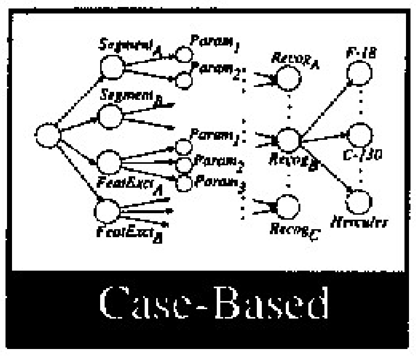
Proceedings

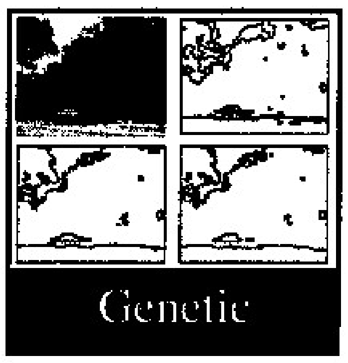
Hyatt Regency Hotel
13 - 16 November
Monterey, California

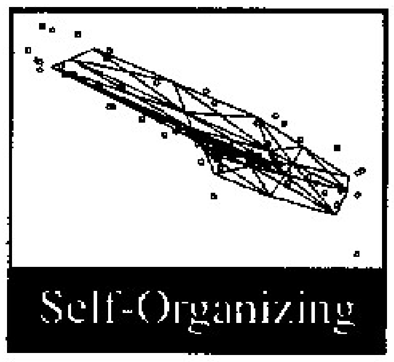
Volume I











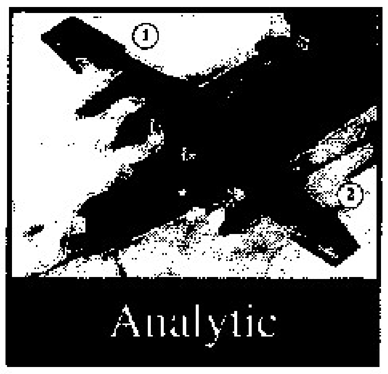


Image Understanding Research Program
University of California at Riverside
Host of 1994 Image Understanding Workshop

Sponsored by:

Advanced Research Projects Agency
Software and Intelligent Systems Technology Office

Distributed by:
Morgan Kaufmann Publishers, Inc.
340 Pine Street 6th Floor
San Francisco, CA 94104-3205
ISBN 1-55860-338-7
Printed in the United States of America

Printed by ***OMNIPRESS . . . "The Proceedings Printer"®***
Madison, Wisconsin (800) 828-0305

TABLE OF CONTENTS

Volume I

Section II - Benchmarks

Section III - RADIUS

Section IV - Image Understanding Environment (IUE)

Section V - Reconnaissance, Surveillance, and Target Acquisition (RSTA)

Section VI - Computers for Image Understanding

Section VII - Sensors for Vision

Section VIII - Vision Architectures

Section IX - Computer Vision Performance Characterization

Section X - Visual Learning

PROGRESS ON VISION THROUGH LEARNING AT GEORGE MASON UNIVERSITY

J. W. Bala, R. S. Michalski and P. W. Pachowicz

Center for Machine Learning and Inference
George Mason University
Fairfax, VA 22030

Abstract

This report briefly reviews the progress of research on learning in vision conducted at the GMU Center for Machine Learning and Inference, in collaboration with the Computer Vision Laboratory at the University of Maryland. The report describes research goals, methodologies, developed systems, and results of applications to selected vision problems. Significant progress has been made in several areas:

(i) Application of symbolic learning and highly nonlinear operators to constructing image descriptions (MLT project)
(ii) Development of a methodology for multistrategy learning which integrates symbolic and neural network learning (AQ-ANN project)
(iii) Learning new concepts by relating them to previously learned concepts (PRAX system)
(iv) Coping with noise in images by an iterative model-driven "detect and purge" method (AQ-NT system)
(v) Adapting to changes in object appearance by incrementally evolving object descriptions (CHAMELEON project).

The developed systems have been experimentally applied to problems of scene segmentation, blasting caps recognition, classification of a large number of textures, and natural object recognition.

This research was supported in part by the Advanced Research Projects Agency under grants F49620-92-J-0549, administered by the Air Force Office of Scientific Research, and N00014-91-J-1854, administrated by the Office of Naval Research. It was also supported in part by the Office of Naval Research under grant N00014-91-J-1351, and by the National Science Foundation under grant IRI-9020266.

1 Introduction

This research is concerned with the development of methodologies and experimental vision systems capable of learning descriptions of visual objects, and applying the learned descriptions to efficiently recognize objects in a scene.

The underlying motivation is that vision systems need learning capabilities in order to be more easily adaptable to different vision problems, and more flexible and robust in handling the variability of perceptual conditions. The project represents an interdisciplinary effort to advance the state of the art in computer vision by applying advanced machine learning methods and to provide solutions to problems unsolved by previous vision research.

One of the significant results of our research was a demonstration that learning methods can be successfully applied to problems of low-level vision. Specifically, the results obtained demonstrate that a multistrategy learning approach that combines rule learning and neural net-based learning can be very successful in fast scene segmentation and object detection. Strong impacts of this research are expected in such domains as industrial object recognition, medical image analysis, sonar-based material inspection, and satellite image interpretation.

2 General Methodology

A "multilevel logical template" (MLT) methodology has been developed for training a vision system to perform a given set of vision tasks. The methodology, developed by Michalski and implemented by Bala, consists of three phases: 1) image marking, 2) automated model development, and 3) model testing (Figure 1).

In **Phase 1**, an operator selects and classifies samples from a training image that represent visual concepts to be learned (e.g., specific objects, parts of a scene, etc.)

In **Phase 2,** the system iteratively executes the following sequence of modules: Training Input Formulation, Model Learning and Refinement and Model Testing.

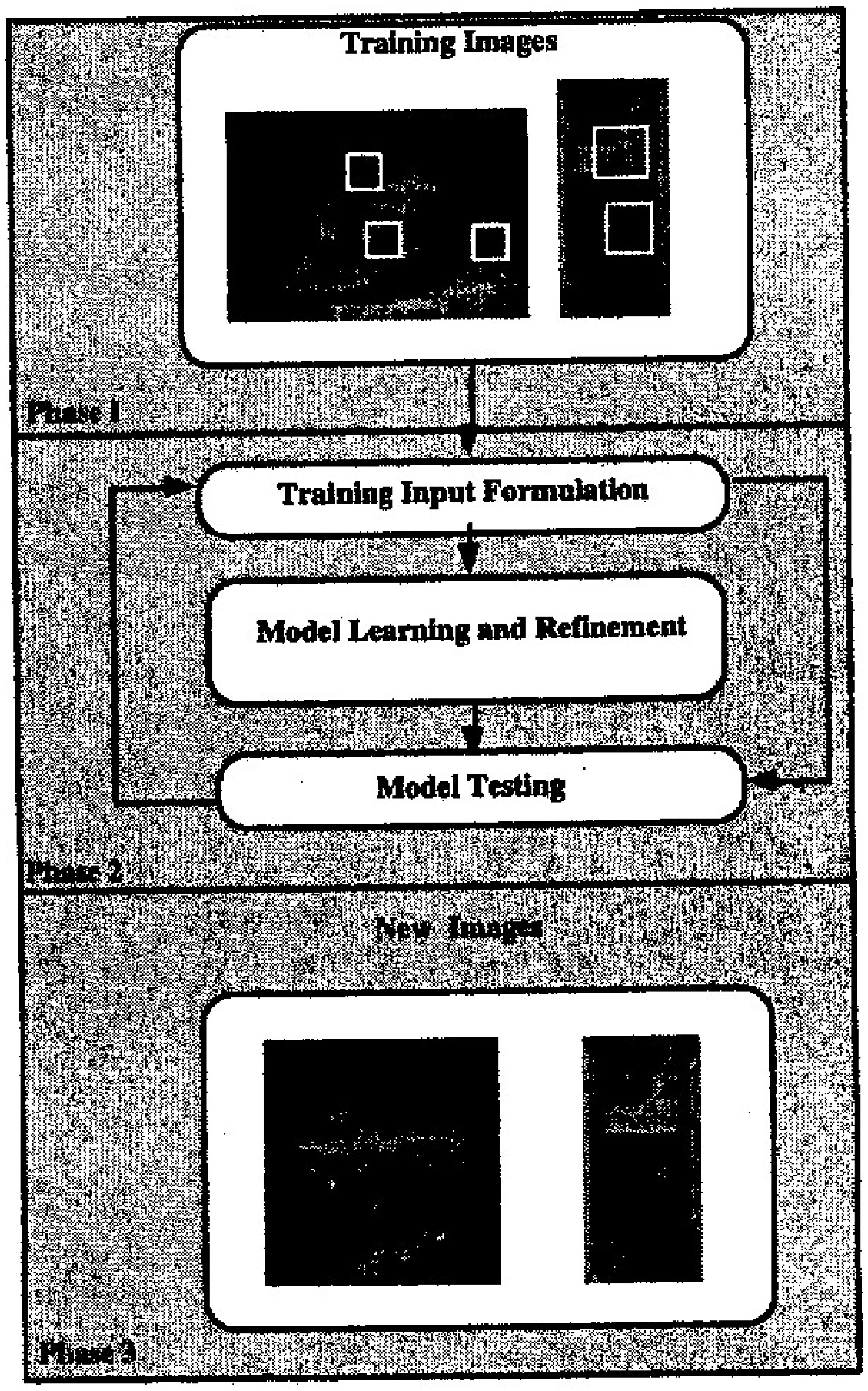


Figure 1: General Methodology.

The **Training Input Formulation** module performs two basic steps: 1) optimizing the image volume (by adjusting the resolution and the number of gray levels accordingly to the given vision task), 2) computing high-level features from the training image samples, and 3) creating "training events," which constitute input to the learning process. The **Model Learning and Refinement** module executes a learning system to determine general descriptions of indicated visual concepts from the given samples (and background knowledge).

At each iteration, the generated descriptions are applied to the whole training area of the image and a "symbolic" image is created, in which the "pixels" denote numerical labels of the visual concepts being learned. The descriptions are called "logical templates," because in the original implementation of the methodology they were logic-style decision rules that will be applied to the image in parallel.

The **Model Evaluation** module evaluates the quality of the descriptions obtained at a given iteration by relating the symbolic images they produce to the target image. If the descriptions need further improvement, the process is repeated as the current symbolic image is input. The process ends when the obtained symbolic image is sufficiently close to the target image labeling (indicating the "correct" labeling of the image). Complete object descriptions are sequences of image transformation operators (rule sets) that produce the output image, and serve as symbolic object models.

Phase 3 involves an application of the learned models to new images, to compute confidence scores for recognition.

To recognize an unknown surface sample, the system matches it with candidate surface descriptions. This is done by applying decision rules to the events in the sample. For each event, the class membership is determined. To increase the confidence of recognition, the majority class of the events in a window is taken as the decision.

Advantages of this approach are that the recognition process can be very fast, as it is amenable to parallel execution, and that the recognition accuracy for new images is very high.

The MLT methodology has been initially applied to learning multilevel rules characterizing given surface classes from surface samples [Michalski et. al., 1993]. The rules were determined using the inductive learning program AQ-15 [Michalski et al., 1986] and represented in the VL_1 logic-style language (Variable-Valued Logic System 1) [Michalski, 1972]. These rules serve as "logical templates" that can be matched in parallel or sequentially against window-size samples of surface to classify the image.

The methodology has been subsequently implemented using different learning systems, which are suitable for different vision problems.

The following learning methods have been included:

(i) Learning symbolic image transformations using the AQ inductive rule learning program AQ15c, [Michalski et al., 1986; Wnek, 1994].

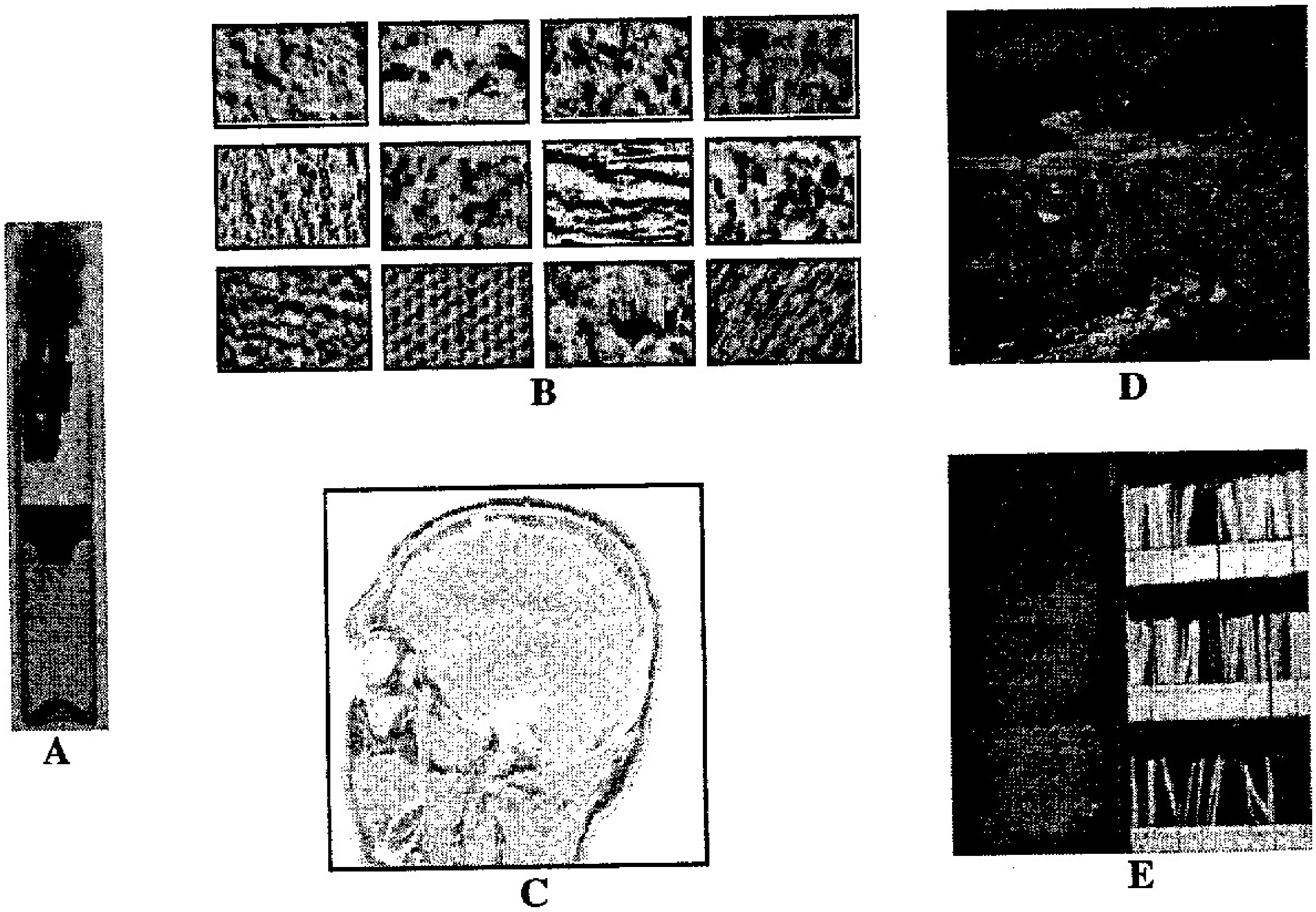
(ii) Multistrategy learning that combines decision rule learning with neural net learning.

(iii) Multistrategy learning that combines decision rule learning with a genetic algorithm

(iv) Class similarity-based learning for building descriptions of large numbers of classes.

The methodology has been applied to several vision tasks: A) Detection of objects belonging to specific classes. Initial experiments have been performed on detecting blasting caps in X-ray images B) Quickly learning to identify textures from a large class of possibilities C) Identification of problem areas in medical images D) Learning to segment natural scenes into concept areas E) Learning to identify objects in indoor scenes (Figure 2).

The next section describes in greater detail individual projects and the results obtained.



Data Set	Vision Task
A. X-ray images of blasting caps	Detection of objects of a specific class
B. Textures	Classification of surfaces
C. MRI images	Analysis of medical images
D. Alpine images	Natural object recognition
E. Indoor scene images	Recognition of indoor objects

Figure 2: Application areas.

3. Research Projects

3.1 Learning Vision Tasks by Combining Symbolic and Neural Network Learning: AQ-ANN

This project, conducted jointly by Bala and Michalski, concerns the development of a new multistrategy learning methodology that is specifically oriented toward vision learning. The methodology combines symbolic rule learning and neural-based learning strategies in order to achieve high efficiency and accuracy in learning object descriptions, and in applying them to object recognition. The core idea is to learn symbolically approximate decision rules for the task at hand, and then use the rules to structure a neural net.

The initially developed vision system has several advantages: it can be easily modified and applied to new problems (due to learning), its learning speed can be at least an order of magnitude faster than neural net learning (due to "symbolic pre-structuring" of the net), it has short recognition times (due to its parallel architecture), and its underlying recognition rules are easy to understand by a human operator (due to the symbolic knowledge representation of the basic decision rules). The developed system was experimentally applied to natural scene recognition.

The method works in two stages. In the first stage, a set of decision rules in the VL_1 (Variable-valued Logic System 1) which approximately characterize objects of interest are induced from examples. In the second stage, the rules are transformed into an equivalent neural net, and the resulting neural net is further trained to improve its recognition performance.

The primary motivation in applying this approach is to increase the execution speed of the recognition system. Another motivation is to represent visual knowledge embodied in the network in an explicit form of understandable rules in order to enable the network's decisions to be understood by humans. The main area of applicability of the developed system is learning high level visual concepts of surfaces in 2-D images (e.g., trees, bushes, bookshelves, cancer cells, etc.).

The AQ-ANN approach showed that a symbolic learning method augmented by parallelism can be successfully applied within time constraints to complex domains like outdoor scene recognition.

The approach combines the well-known AQ algorithm for rule learning with standard neural net learning (hence the AQ-ANN name for the project). The AQ algorithm generates decision rules in a "greedy" fashion, at each step determining one rule that covers a maximal portion of the "uncovered" training data, and so on until all positive training examples are covered, and all negative examples are excluded. To create rules from examples, it employs "inductive generalization operators" that make the decision rules as general as possible without becoming inconsistent [Michalski, 1972; Michalski et al., 1986]. When noise is present in the training data, the rules are allowed to be partially inconsistent and/or incomplete with regard to the input data.

The learning process is executed in two phases:

1. Rule learning using the AQ algorithm.
This phase generates rules that describe the training examples (those that cover only a few examples are truncated from class description).

2. Backpropagation network learning.
Each node in a one-layer network corresponds to a single rule. The degree of match of an example to the node rule represents node activation. This activation value is input to the sigmoid transfer function associated with each node. Weight values for the connections between nodes and outputs are obtained using the backpropagation learning mechanism.

The node rules in the network are a form of receptive field transfer function. The network architecture is similar to the Radial Basis Function network (RBF network). The RBF network models data by a Gaussian distribution function associated with each node. The network generated by the AQ algorithm is constructed based on rules that represent generalization of the initial examples. Our approach overcomes two important drawbacks of RBF learning algorithms, namely, choosing the right number of nodes (clusters to be modeled by the Gaussian distribution) and the measure of the spread of the data associated with each cluster.

Figure 3 illustrates an application of the AQ-ANN method to the problem of learning three concept classes ("Tree area", "Grass area" and "Rock area") in "Alpine images." There were five attributes computed for each pixel of image section. First two attributes represented detection of horizontal and vertical lines in a 5 by window. The remaining three attributes represented color intensity of Red, Green and Blue composites. The top of Figure 3 shows the training scene, and training samples selected from it. The bottom left image shows a new scene to be segmented into the above three concept classes (the image has been quantized into 98304 pixels). The "Target" shows the "ideal" segmentation of the image done by a human operator. The "Result" presents the image segmented by the neural network structured by the AQ-15 learning program. It shows the final segmentation that was obtained by substituting class membership for each pixel based on a class majority in a 15 by 15 window. The "Result" image shows 100% correct recognition for most areas of the new scene. Since the target image labelling performed by a human operator is imperfect (small areas of class "Rock" are not shown) the recognition rates approximate the correct rates.

It can been seen that the "Result" image and the "Target" image are very similar. Table 1 presents a comparison of learning and recognition times, and performance accuracy in this experiment for three learning approaches: AQ, AQ-ANN (neural network structured by the AQ program), and a direct application of the neural network (with binary coding of attribute values and backpropagation network learning). The architecture was simulated on the MATLAB neural network toolbox, and was able to process the whole scene in about 20 seconds of CPU time on a SunSparc 2 workstation. As indicated in Table 1, by applying the multistrategy AQ-ANN approach, the image recognition time was reduced from 500s to 20s. The network architecture structured by the AQ-ANN system is shown in Figure 4.

Compared to symbolic learning the learning time for the neural network was three orders of magnitude slower, but it was faster, by one order of magnitude, in recognition time. As shown in Table 1 the highest recognition accuracy has been achieved by a multistrategy learning, AQ-ANN method. At the same time the learning time for AQ-ANN was 50 times faster than for the 2 layer neural net.

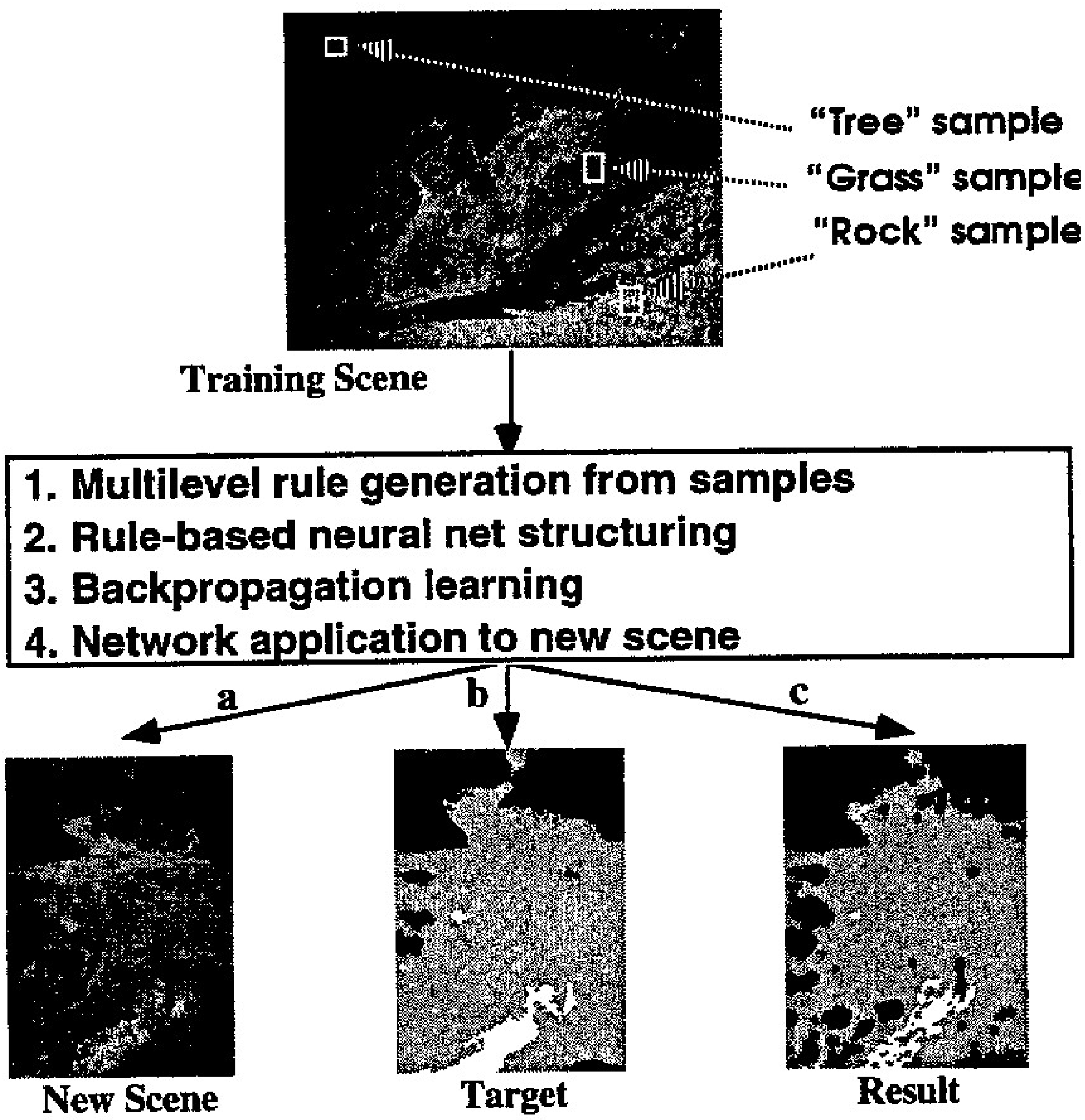


Figure 3: An illustration of the application of the AQ-ANN method to learning three visual concepts, "Tree area", "Grass area" and "Rock area," from training samples.

Learning Approach	Learning time (CPU time in sec.)	Recognition time (CPU time in sec.)	Recognition accuracy pixel-based	Recognition accuracy window based
Symbolic learning: AQ-15 program	6	500	88%	~100%
Neural network backpropagation learning: 1 layer network	no convergence	N/A	N/A	N/A
Neural network backpropagation learning 2 layers	5690	25	85%	not computed
Multistrategy learning: AQ-ANN 1 layer structured by the AQ-15 program	120	20	92%	~100%

Table 1: Comparison of learning/recognition times, and recognition rates for symbolic learning only, neural nets learning only, and integrated symbolic and artificial neural net learning.

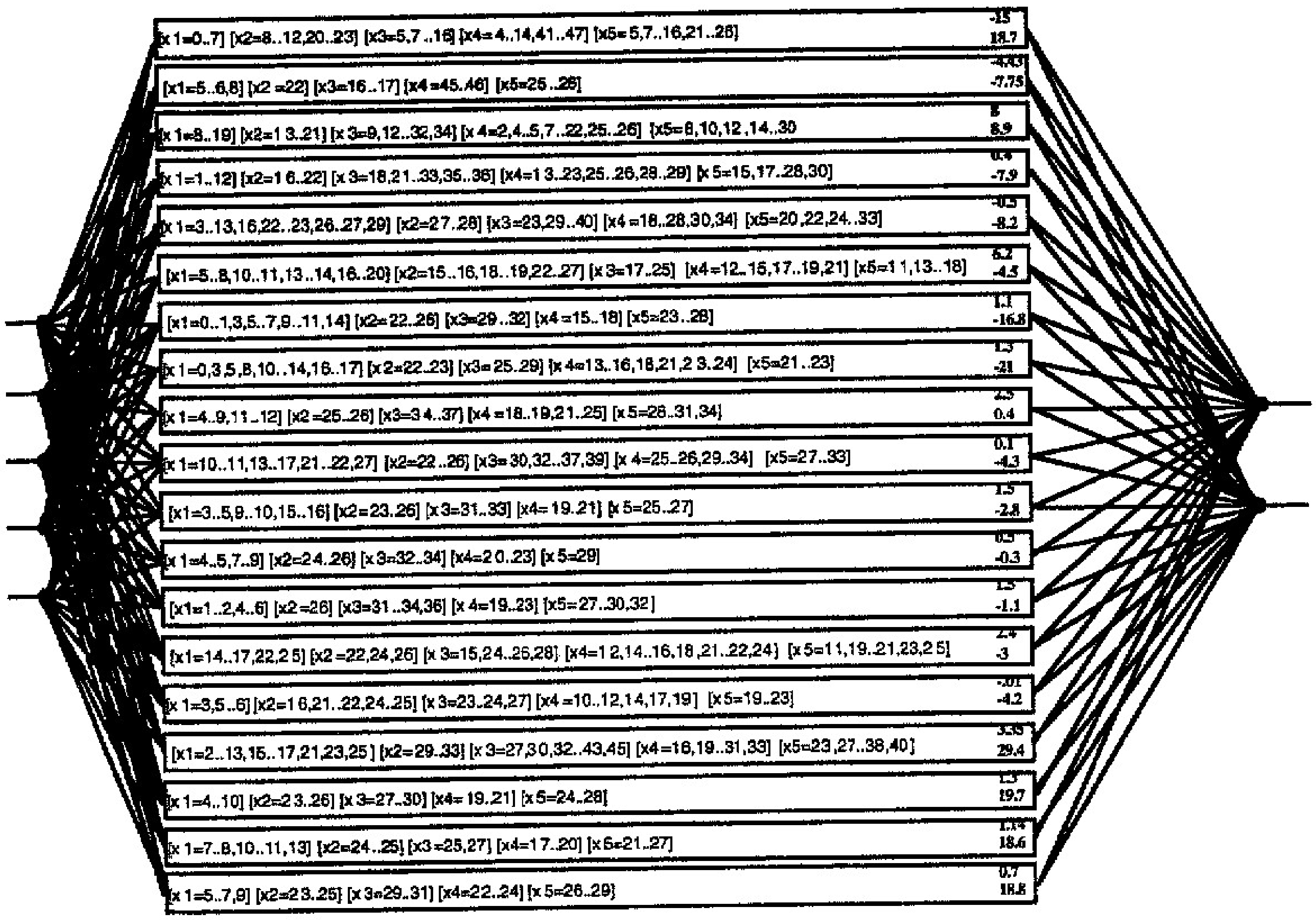


Figure 4: A neural network structured by rules learned by the symbolic AQ-15 program.

3.2 Learning to Recognize Large Numbers of Classes (PRAX-2)

Most research on concept learning from examples concentrates on algorithms for generating concept descriptions of a relatively small number of classes. In conventional methods, when the number of classes grows, their descriptions become increasingly complex, in order to discriminate each class from the previous classes.

In some applications, the number of classes may be very large, and they may not be known entirely in advance. In such situations, the learning method must be able to incrementally learn new classes. Such a *class-incremental* mode is different from the conventional *event-incremental* mode, in which examples of classes are supplied incrementally, but the set of classes remains unchanged.

The PRAX approach was developed for learning descriptions of a large number of classes in a class-incremental mode [Bala et al., 1992]. The learning process consists of two phases. In Phase 1, symbolic descriptions of a selected subset of classes, called *pr*incipal ***axes*** (briefly, *praxes*) are learned from concept examples (here, samples of textures). The descriptions are expressed as sets of rules. In Phase 2, the system incrementally learns descriptions of other classes (*non-prax* classes). These descriptions are expressed in terms of their similarities to praxes, and thus the second phase represents a form of analogical learning. To utilize a uniform representation, the prax descriptions are also transformed into sets of similarities to the original symbolic descriptions.

PRAX-2 (Figure 5) extends the initial PRAX method by making it more efficient [Bala, et. al., 1993]. This is accomplished by reducing the number of PXs in the changed representation. The selection or deletion of a given PX is based on its discriminatory power, measured as the standard deviation of its values through all classes. In experiments with 24 texture classes (100 training examples per class and 100 testing examples per class) the number of PXs generated from the initial 8 classes was reduced from 170 to 17. Thus, all 24 classes were recognized using only 17 PXs (rules). Figure 6 shows examples of a PX expressed as a conjunction of attribute conditions and a class description (SV) expressed as a vector of 17 similarity measures.

The ability of the method to describe many classes while using only a small set of rules has been shown to be very promising in initial experiments. The main strength of the method lies in a problem-relevant transformation of the description space. The new descriptors form generalized sub-spaces of the initial training space.

Given:

P - set of principal axes found by PRAX {PX1, PX2, ..., PXM}
M - number of principal axes
K - number of training classes
ϑ - maximum discriminant standard deviation

Do:
For each $PX_i \in P$
    For each class k
        Calculate average match to examples of class k (S_{ik})
For each $PX_i \in P$
    Calculate a standard deviation $\sigma_i = \sigma\{S_{ik},\ k=1,.., K\}$

For each $PX_i \in P$
    If $\sigma_i < \vartheta$ then remove PX_i from P

Figure 5: Algorithm for determining principal axes.

PX == rule =>[x1=8..21] & [x3=15..22] & [x4=24..42] & [x5=19..37] & [x6=28..36] & [x7=28..36] & [x8=12..25]

SV(class C16) => [5.1, 57, 21, 51, 41.2, 6.2, 0.4, 6.3, 0.4, 9.6, 95, 94, 89, 93, 87, 28, 0.3]

Values in the SV vector represent a normalized (range 0 to 100) average match of examples of the C16 class to 17 PXs

Figure 6: Examples of a PX and a class description.

3.3 Noise-Tolerant Learning of Object Models from Complex Sensory Data

This project is directed by Pachowicz and aims at the development of new techniques for learning from very complex and noisy attributional data. The guiding premise of this research is that erroneous data can be detected more effectively on the model level — where relationships between data clusters and between classes to be learned is expressed better than in raw training data. These techniques are dedicated for symbolic learning programs, however, we are also adapting them to the other classifiers.

Model acquisition from noisy data sets is a difficult problem for symbolic learning programs. Inductive learning systems perform a generalization of the input data in order to anticipate unseen examples. In a standard mode, when all the input examples can be assumed to be correct, a concept description generated by an inductive learning system should be complete (cover all training examples) and consistent (cover no examples of other concepts). In the case of noisy data, the system does not seek such complete and consistent descriptions.

There are two basic approaches to symbolic learning from noisy data. The first approach, tree pruning (elimination of some subtrees from the learned decision tree), taken by the ID family of algorithms, allows a certain degree of inconsistent classification of training examples so that the descriptions will be general enough to describe the basic characteristics of a concept. The second approach, taken by the AQ family of programs, is to remove some of the unimportant rules (or conditions) from a set of rules, and retain only those covering the largest number of examples. Traditional learning methods based on pruning/truncation try to handle noise in one step. Therefore, they share a common problem: the final concept descriptions are based on the initial noisy training data.

A new approach has been proposed which extends the traditional one-step method of noise handling to a closed-loop two- or multiple-step process. The learning loop is presented in Figure 7. It includes: (1) concept acquisition by a concept learning system such as AQ or ID; (2) evaluation of learned class descriptions, detection of less significant disjuncts/subtrees, which are not likely to represent patterns in the training data, and removal of detected rules/subtrees; and (3) filtration of training data through optimized rules/trees (i.e., removal of all examples not covered by truncated or pruned concept descriptions). This learning loop can be run once or multiple times with changing learning and/or truncation/pruning criteria.

In this approach, pruned/truncated concept descriptions are used as a filter to improve the training data set. Then, the concept acquisition phase is repeated from the improved training data. Consequently, those training examples which caused the generation of pruned/truncated concept components are no longer taken into account when concept learning is repeated. Since the detection of erroneous examples is executed on the concept description level rather than on the input data level, data filtration reflects attribute combination in the construction of concept descriptions and inter-class distribution over the attribute space.

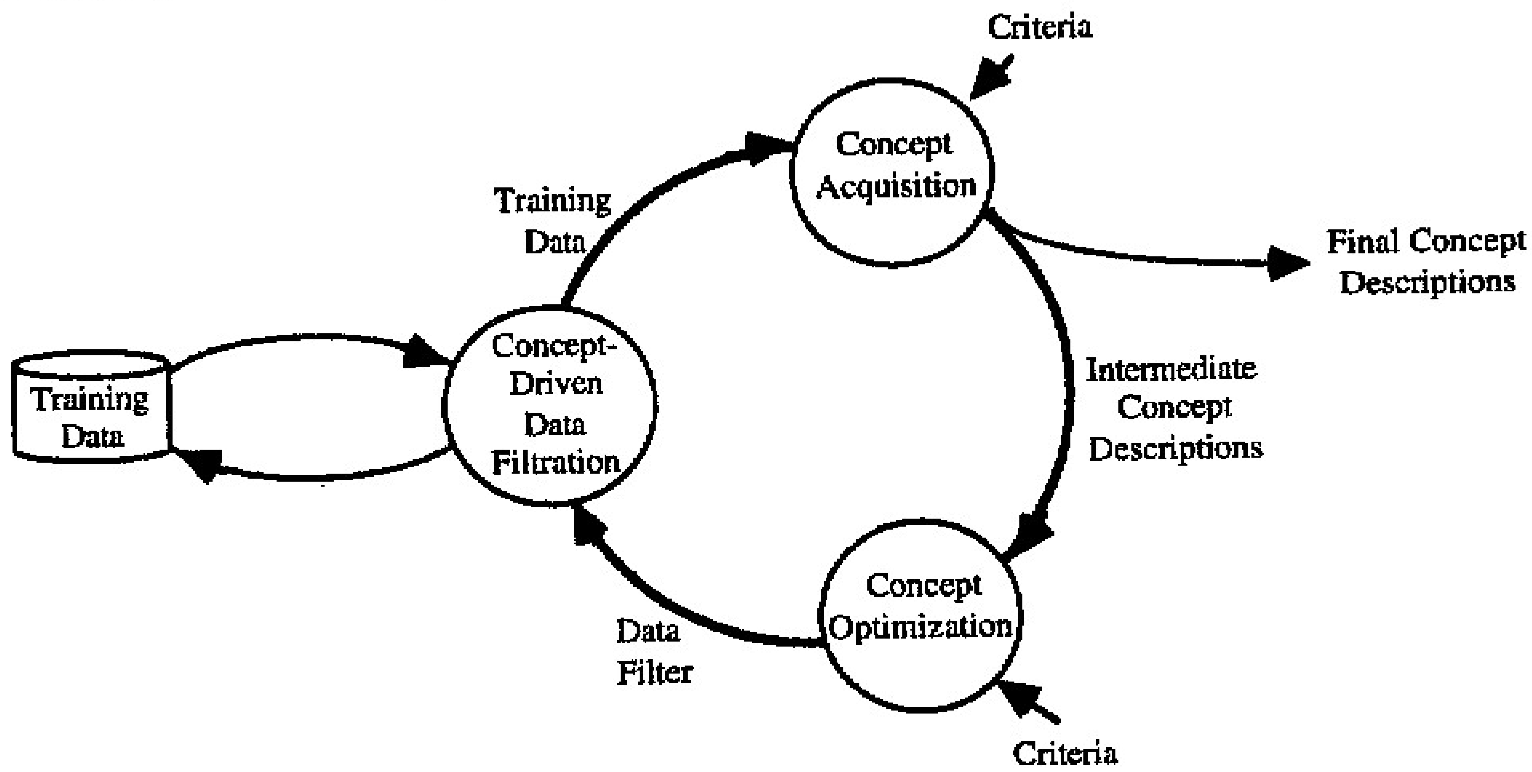


Figure 7: Learning loop.

Previously, we prototyped an introductory version of a rule learning program and showed basic results for a simple texture recognition problem involving six texture classes. We reported that the recognition rate increased and the complexity of object models decreased substantially. Recently, we implemented the above approach to rule learning and decision tree learning programs and tested them on several vision problems [Bala and Pachowicz, 1993; Pachowicz and Bala, 1994b]. The new version of the learning program AQ-NT uses the AQ14 [Michalski, 1985] learning program. The decision tree version, the ID-NT program, uses the C4.5 [Quinlan, 1993] learning program. Both programs were tested on the acquisition of attributional descriptions of twelve similar texture classes from texture energy measures. Different image sections were used for training and for testing.

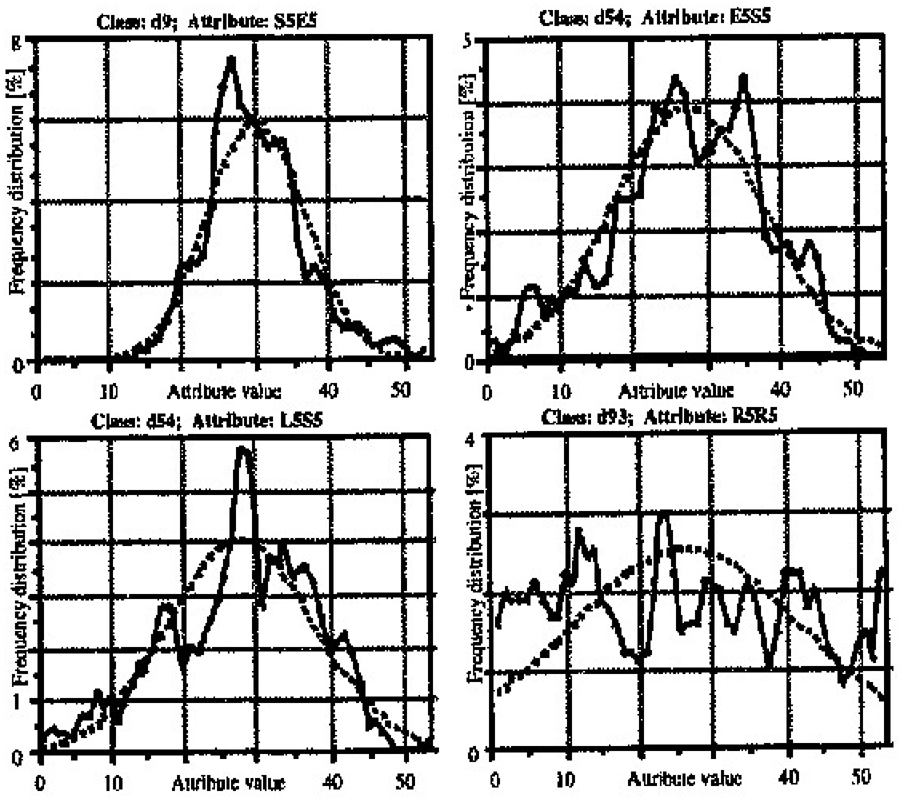


Figure 8: Example attribute value distributions for the texture data.

Figure 8 presents selected examples of attribute value distributions (the most complex distributions) for individual classes, where the solid line corresponds to a smoothed attribute distribution, and the dotted line corresponds to the approximated normal distribution. In some cases, the distribution was multi-modal. Sometimes an attribute had a uniform distribution for a single class, but was very distinctive for the remaining classes.

The recognition results for the AQ-NT program are presented in Figure 9 [Pachowicz and Bala, 1994b]. The average error rate over twelve classes decreased from 29.3% to 28% in the range of truncation levels from 0% to 10%. The truncation level corresponds to the number of training data covered by truncated components of concept description. At the same time, the standard deviation from the average error rate decreased from above 25.5 to 24. Most importantly, the maximum error rate computed over individual classes (this rate corresponds to the worst recognizable class) decreased from above 65% to below 59%. For higher truncation levels the maximum error rate stabilized. This result has been found very encouraging because it improves the recognition of the worst recognizable class.

The experimental results obtained with the ID-NT program are presented in Figure 10 (white marks) along with the results for the original C4.5 learning program (black marks). The average error rate decreased from 34% to below 33.5% along with a decrease in the standard deviation. Most importantly, the maximum error rate for individual texture classes decreased very significantly, from 70% to 60%; this indicates better selection of the first attribute in tree generation, which in effect improves the recognition of the worst recognizable class.

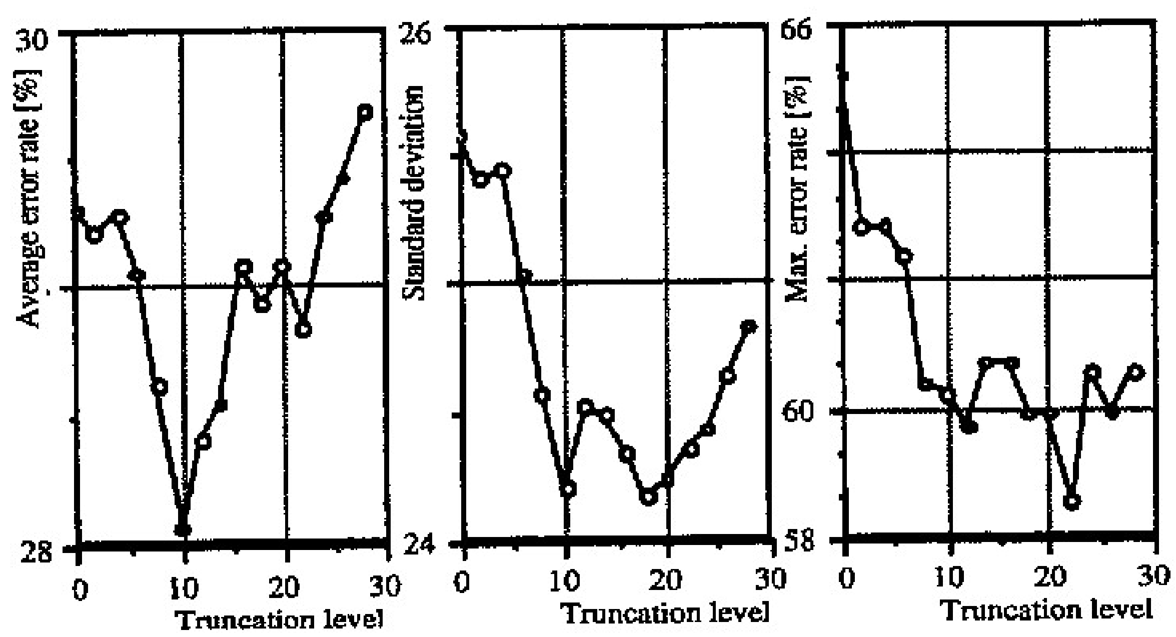


Figure 9: Recognition results obtained by the AQ14-NT noise-tolerant learning program.

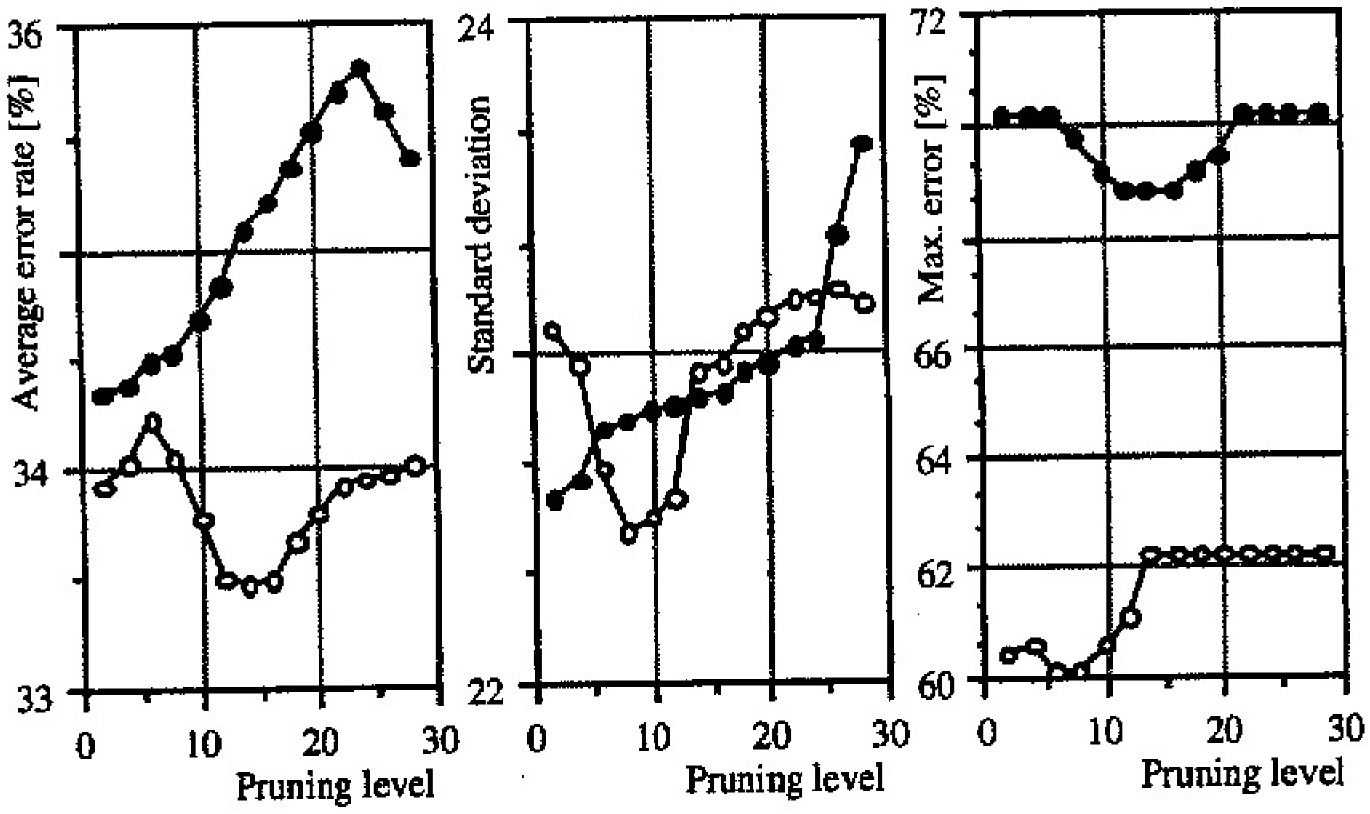
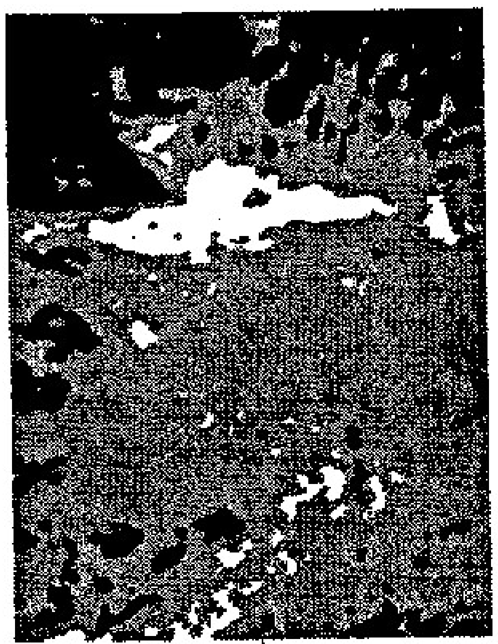


Figure 10: Recognition results for the C4.5-NT noise-tolerant learning program (white marks) and for the C4.5 learning program (black marks).



a)

b)

Figure 11: Classification results for surface class descriptions learned by: (a) the AQ14-TRUNC program, and (b) the AQ14-NT learning program.

The developed noise-tolerant learning method was tested on the real images of natural outdoor scenes shown in Figure 3. All the images were taken in different places but in the same mountain area. There were difficulties with the precise segmentation of the test image because of (i) the lack of a clear border area between the "Grass" area and the "Tree" area, (ii) many isolated large rocks, (iii) overlap of the "Grass" area and the "Rocks" area, and (iv) difficulty in the interpretation of some small image region.

The results of model based scene segmentation are presented in Figure 11. There is a significant improvement in the classification results when the class descriptions were acquired by the AQ-NT program. There are two major improvements. First, the distinction between the "Tree" area and the "Grass" area is improved. Second, the false classification of large grass sections is eliminated. Moreover, the picture better highlights surface details corresponding to large rocks and small bushes.

3.4 Learning Descriptions of 2D Shape

For many visual concepts, the shape of an object is its primary distinguishing property. Often, shape information alone is sufficient for object recognition (e.g., to recognize a generic face). In other situations, shape has little value, but surface information is important (e.g., to recognize a type of material). Thus, depending on the task at hand, a vision system can use shape information alone, surface information (texture/color) alone, or a combination of both to recognize objects or discriminate among classes of objects.

This project, conducted by Maloof in cooperation with Michalski, is concerned with the application of a machine learning system to shape recognition. We have applied the AQ-15c symbolic learning system [Bloedorn et al., 1993; Wnek, 1994] to learn shape descriptions of x-rays of blasting caps, and have compared the prediction accuracy (the ability to recognize unseen examples) of our results to the K-nearest neighbor method [Weiss and Kulikowski, 1992], a statistical pattern recognition technique, and to backpropagation neural networks [Zurada, 1992], a non-symbolic machine learning method. Our approach consisted of a five-step learning and recognition methodology:

1. Image Data Reduction
2. Blob Isolation
3. Event Extraction
4. Learning
5. Recognition

Image data reduction is designed to eliminate extraneous image data while preserving pertinent information crucial for object recognition. Essentially, this is an abstraction procedure. The image volume was reduced by scaling along the X and Y dimensions and by quantizing the image gray levels. This process was guided by information-theoretic measures of the information content of the extracted events. The original image set was taken from an image database and consisted of 25 x-ray images of blasting caps

Blob isolation involves localizing characteristic image regions, or blobs, using traditional threshold operations, following an approach similar to Sydow and Cooper [1992]. The threshold level was determined by histogram statistics and was used to isolate three characteristic blobs that served as classes for the learning algorithm. These blobs corresponded to high-density structural features present in the blasting caps. Not all structural features were present in all blasting caps.

Once isolated, fourteen statistics were calculated from each blob in each image. The statistics included area of the region, perimeter around the region, average gray scale value within the region, and the like. Each set of blob statistics served as an event, or concept example. Event extraction produced 66 events which were partitioned into three classes corresponding to the characteristic image blobs. Typically, symbolic learning algorithms require discrete attributes. Consequently, the ChiMerge algorithm [Kerber 1992] was used to scale the real-valued statistics into discrete linear attributes, each having between 10 and 15 value levels. The ChiMerge algorithm groups real-valued attributes into discrete intervals based on statistical measures of similarity. Lastly, the extracted events were partitioned into training and testing data sets as prescribed by a 3-fold cross-validation methodology [Weiss and Kulikowski 1992].

Symbolic learning was performed using the AQ-15c attributional learning system which induced characteristic descriptions of each blob for a given training set. On average, AQ generated one description, or rule, for each class. The rules consisted of fourteen conditions which corresponded to the fourteen extracted statistics.

The final stage was to use the learned descriptions for recognition. Events separated for testing are classified using the learned descriptions and the classification accuracy is calculated. For comparison, recognition accuracy was also computed using a K-nn classifier and a backpropagation neural network. Average classification accuracies over three trials are presented in Table 2.

Learning Method (Blasting cups)	Classification Accuracy
AQ-15c	97.22%
K-nn	95.71%
Backpropagation NN	95.71%

Table 2: Performance summary for classification technique.

Although the classification accuracies of these classifiers are similar, AQ-15c has advantages over the other two methods. First, AQ produces symbolic, declarative descriptions. Unlike neural net learning, these AQ-generated descriptions can be easily interpreted by human operators, as they represent concepts both intuitively and literally. Unlike the K-nn method, the AQ rules can be executed in parallel, which is important for fast recognition. With backpropagation, concept descriptions are distributed throughout the network structure as connection weights. For sufficiently complex networks, it is impossible to directly understand what was learned. With K-nn, the classification of an unknown example is made by measuring the distance between the unknown and K of its nearest neighbors. Consequently, our only feedback is a real-valued distance. In fact, with both K-nn and backpropagation, we cannot know which attributes are most relevant for classification without additional statistical analysis.

In addition, because AQ descriptions are literal and symbolic, we can optimize these descriptions, either manually or automatically, to yield higher classification accuracy. For the reasons discussed previously, K-nn and backpropagation concept descriptions cannot be inspected, much less optimized.

Since this is early work, much work remains. We are currently investigating methods of optimizing the representation space using information-theoretic measures; that is, using information-theoretic measures on the extracted statistics to determine which image reduction parameters produce the most discriminatory features. Future work involves learning invariant shape characteristics, in particular to scaling, rotation, and other class-preserving transformations. Further, we want to investigate whether the ChiMerge algorithm, a necessary pre- processing step for AQ, affected the recognition rates of K-nn and backpropagation.

3.5 Dynamic Determination of Key Features for Object Recognition

This project, conducted jointly by Hadjarian and Michalski, aims at developing new methods for recognizing objects from a large set of possible objects. This is to be done without matching rules/models, but by dynamically determined "key" characteristics. The underlying assumption for this method is that to recognize objects in a given context only partial information is usually sufficient. For example, to differentiate just between cucumbers and bananas, it might enough to know only the color of the objects. Thus, just by determining the value of one attribute, color, the system may be able to recognize the object as one of a cucumber or a banana.

As the number of classes of objects that the system is supposed to recognize increases, so does the number of features necessary to achieve the correct recognition. For example, let us assume that the recognition system mentioned above should also be capable of recognizing lemons in addition to cucumbers and bananas. It is easy to see that color by itself is no longer sufficient for distinguishing between these fruits since bananas and lemons can both be yellow. Thus such a system needs to extract other features such as shape or texture information in order to achieve its classification task. The feature selection problem addresses the issue of finding features which are sufficient for the given classification task. However, a large number of classes usually means a large number of features.

In order for an object recognition system to recognize an instance of an object, it needs to store a model of that object in its database. This model contains all the important features of the object. An image is recognized as an instance of the object if there is a match between the features extracted from the image and the features stored in the model. In most object recognition systems, feature extraction and classification are two isolated processes. The feature extraction module first extracts all the relevant features of the image which are necessary for achieving correct classifications of all objects which the system is trying to recognize. The classifier will then classify the image by comparing theses extracted features to those from the models stored in the database.

The disadvantage of such a system is that in order to recognize an object, it needs to always measure the same properties of it, namely all its relevant features. This is, however, not desirable since extracting all the relevant features can be computationally very expensive and is not always possible. This is especially true for a system which recognizes a large number of objects, since as mentioned earlier this usually requires extraction of a large number of features.

We are proposing an alternative approach to recognition. The idea is based on Michalski's Theory of Dynamic Recognition which was originally introduced in 1986. The main idea behind dynamic recognition is that the system determines "key" attributes from *characteristic descriptions* of objects. These attributes are determined by conducting inductive inference on candidate object descriptions.

The proposed Dynamic Recognition approach involves three steps:

1- REDUCE
2- INDUCE
3- INQUIRE

In the REDUCE step, some "striking features" of objects in the image are used to reduce existing characteristic descriptions and determine candidates. In other words, all the rules which are not satisfied by the values of these features are removed from the set of candidate descriptions. In the INDUCE step the AQ program is applied to the reduced set of characteristic descriptions to determine the simplest *discriminant* recognition rules. These discriminant rules will usually contain only the discriminant features, i.e. fewer features than the original characteristic descriptions. In the INQUIRE step, an evaluation function is applied to each remaining feature and the value of the feature with the highest score is extracted from the image of the object to be recognized. An important parameter of this evaluation function is the *cost* of the feature, which measures the difficulty of extracting it from the image. Rules not satisfied by the value of the extracted feature are removed from the set of candidate descriptions. The INQUIRE step is repeated until we are left with one candidate description, namely the description of the object in the image.

Thus, recognition is considered as an inductive inference process that determines the discriminant features of the objects in a given context, and not as a matching process.

3.6 Model Evolution Paradigm to Object Recognition in Dynamic Environments

This project is directed by Pachowicz and aims at object recognition under the gradual change in perceptual conditions and/or under varying object appearances.

Most research on object recognition has been focused on learning to recognize objects under a given subset of stationary perceptual conditions (such as lighting, resolution and positioning) and for known object appearances (e.g., subsets of IR or SAR object signatures; subsets of object silhouettes). Object recognition in dynamic environments, however, has to deal with changes in perceptual conditions and object appearances not known to the system beforehand. Frequently, models learned under given perceptual conditions are not effective in recognizing objects under other conditions. This problem is particularly severe for object recognition in outdoor environments where the variability of perceptual conditions and object appearances can be extremely high.

Most approaches to object recognition do not adapt an object recognition system directly to changing perceptual conditions and object appearances. These methods use stationary models acquired during the off-line training phase. Such an approach requires that each condition influencing the change of object characteristics is represented in the model, a conclusion which is hard to satisfy for realistic environments.

We have developed a model evolution paradigm for object recognition under variable perceptual conditions and changing object appearances. The paradigm relies on the on-line dynamic modification of object models according to perceived changes in object characteristics. This paradigm was tested for a scene segmentation problem based on texture characteristics of surfaces. It assumes that a change in, for example, texture characteristics is gradual and is reflected in the images of a sequence. Given texture descriptions (models) learned from the first image of a sequence, the system applies these descriptions to the next image to recognize the objects. Then, the system computes a recognition confidence for each object and compares the results with those obtained when working with the previous images. Dynamic characteristics of the confidence change are modeled. If the recognition confidence deteriorates, so that the system will have more problems in recognizing the object in the next image, the system indicates which descriptions must be modified and activates data selection and learning processes. New training examples, which represent the change in object characteristics, are selected and provided to an incremental learning program. The modified models are verified to insure the soundness of the evolution process.

Using the model evolution paradigm, a vision system adapts to the changes in the environment by adapting the object models on-line and autonomously. This allows for capturing any variability in object characteristics without knowledge about object properties and without building complex, dedicated modules to deal with changes in a given perceptual condition. Thus, an object model can be adapted to any combination of perceptual conditions. Moreover, the system can adapt to a change in the internal state of an object (e.g., to a change in a target's IR signature). Model evolution is an active agent process actively working on its internal knowledge and models of the environment and the objects. Model evolution includes (but is not limited to) and integrates: vision processes, model evaluation, reasoning about the models, guidance for model modification, data selection, and control processes. A kernel of the model evolution system is an incremental learning program.

We have developed the CHAMELEON-1 (semi-autonomous evolution) and CHAMELEON-2 (fully autonomous evolution) systems for the recognition of textures and for texture-based scene segmentation under gradual changes in resolution and lighting conditions [Pachowicz, 1993a, Pachowicz, 1994a]. Recently, the CHAMELEON-1 and -2 systems were intensively tested. We used different image sequences and different control strategies for the selection of new training data for model evolution. We investigated the soundness of the model evolution in critical situations --- i.e., situations where the system mistakenly selects incorrect data or the dynamics of model evolution is too slow when compared to the dynamics of the change in object characteristics. We worked with two incremental learning programs, AQ-14 and AQ-15, as the kernel of the model evolution system.

Conclusions from the development and testing of the CHAMELEON-1 and -2 systems have been used in the design of a new framework for the application of a Bayes classifier and a radial basis function classifier (RBF) to serve as the incremental learning kernel of a model evolution system. The new kernels will be capable of modifying the models more effectively using: (i) statistical information and/or selected new training data, (ii) gradient information about the direction and the dynamics of model change within the attribute space, and (iii) prediction of model change beyond the image sequences already seen. We also investigated (1) architectures for the integration of vision and learning processes of model evolution particularly for automatic model evolution guidance, (2) problems with instability in model evolution, and (3) different strategies for the selection of new training examples for model modification in the incremental mode.

New application domains we are experimenting with include Automatic Target Recognition and segmentation of medical image sequences (brain cross-sections). These applications are characterized by variable target/tissue appearances perceived over time and/or space. The model evolution paradigm for object recognition and image segmentation is particularly useful in those situations because (i) complete models are hard to obtain, (ii) changes in the environment and perceptual conditions significantly influence the object signature, (iii) the system works with image sequences, and (iv) a given image sequence represents a gradual change (rather than a step change) in object characteristics. We investigated the change in ATR data using Lincoln Lab turntable ISAR data of four targets (Camaro, Dodge van, Pickup truck, Bulldozer). Initial results justify the application of model evolution to the ATR problems, for example, to adaptive sensory guidance of a torpedo.

3.7 Autonomous Vision Agents: Learning, Evolving and Self-governing

This new research project is directed by P. Pachowicz. It aims at the design and development of adaptability mechanisms for a vision module which is already prestructured for application-specific data gathering and/or image analysis/understanding. These mechanisms will allow a vision module to undergo on-line modification of its internal knowledge/models, structure and/or processes in an active manner.

This research focuses on how an autonomous vision agent can manage itself while working in dynamic environments, under varying task parameters, and employing dynamic links with associated subsystems. We identify the following basic issues that the agent has to deal with on-line:

(i) change in scene complexity influencing the time, quality and complexity of processes needed for image analysis/understanding,
(ii) change in object appearances, influencing the change of object/scene models,
(iii) occurrence of unexpected situations the system has barely been trained to deal with,
(iv) on-line change in task parameters, and
(v) interruptions/requests coming from processes that the agent communicates with (sensor hardware, host task processes, and application processes).

Sensory systems working in realistic dynamic environments may have to deal with one or more of these issues in order to become autonomous and no longer rely on an engineer to reconfigure the system. An autonomous vision agent must be able to minimize the impact of these issues on its perceptual skills.

The way we have chosen to realize this goal is to develop an active vision agent (AVA) which will be capable of modifying its internal resources over a sequence of images affected by situations which differ from those the system was prestructured for. We have designed a framework for an AVA. This framework includes the following three elements which will insure the system's adaptability to changes in environments, parameters of perceptual tasks, and interactions with the other processes of the application system:

1) introduction of different learning functions into the agent's data processing/analysis algorithms,
2) introduction of model evolution processes into the agent's model/knowledge base, and
3) introduction of self-governing processes into the agent.

The first element of an AVA, learning functions for data analysis algorithms, allows the agent to optimize itself to operate better and faster for repetitive tasks/conditions. Using these functions, the system constantly looks for better data analysis solutions through a network of prestructured/available image analysis procedures. This recently initiated research has shown how the introduction of learning functions within the traditional *train-recognize* paradigm can transform this paradigm into an active agent paradigm [Pachowicz, 1993b].

The second element of an AVA, model evolution, insures system adaptability to changing object appearances and perceptual conditions not reflected in the initial models. We have developed and tested model evolution systems operating in semi-autonomous and fully autonomous modes for scene segmentation and recognition tasks [Pachowicz, 1993a, 1994a].

The third element of an AVA, the self-governing aspect, supports automatic reconfiguration of agent processes due to changes in scene complexity, time restrictions, task parameters, external requests, and dynamics of the environment. This research has roots in our previous work [Pachowicz, 1992] where we showed how a vision system can restructure itself on-line using simple image measures and a feedback control loop. Our recently developed framework for an AVA includes self-governing functions for the agent through the use of the following tools:

(i) *Focus-of-Attention*: allowing for selective analysis of local image data and/or time events,
(ii) *Resolution-on-Demand*: allowing for accessing data at appropriate levels of detail,
(iii) *Abstraction-on-Demand*: allowing for accessing models/knowledge on appropriate levels of competence, and
(iv) *Event-on-Pipeline*: allowing for incremental analysis of scene objects and events over image sequences.

We believe that by introducing this paradigm into machine perception, autonomous vision agents will gain enough degrees of freedom to adapt to changing external influences.

3.8 Learning about the Environment

This project is directed by the team at the UMD Computer Vision Laboratory. The research has been concerned with:

(i) Development of specifications for agents that are capable of performing given tasks in a given environment. This will be done in the context of a general framework for agent and task specification.

(ii) Development of exploratory and computational strategies that can be used by an active agent to discover and organize information about the structure of its environment. This too will be done within a task-dependent framework.

(iii) Definition of methods of sensor-based manipulator control based on perceptual-kinematic maps, which relate properties of the sensory data (e.g., positions of features in an image) to properties of the kinematic chain that drives the manipulator (e.g., joint angles). In this framework, manipulator control can be regarded as a problem of planning paths on a perceptual-kinematic surface.

Research at UMD during the past year has concentrated in areas (ii) and (iii). Some of the results are described in two papers in these Proceedings [Rivlin and Rosenfeld, 1994; Herve, 1994].

5 Summary

The GMU research on machine learning in vision has developed several novel ideas and systems for applying advanced methods of machine learning to vision. Particularly significant progress has been made in such areas as the development of the multistrategy learning methodology that combines symbolic learning with neural net learning. Experiments have shown that this methodology may increase learning speed by an order of magnitude, significantly increase the prediction accuracy of learning, and at the same time facilitate rapid object recognition due to the parallel architecture of neural nets.

We have developed the PRAX system which uses ideas based on analogical learning acquire descriptions of large numbers of classes. To cope with noisy data, we have developed a methodology for noise detection and purging of data, which have shown very promising results.

Further progress has been made on the development of a methodology for learning to recognize objects belonging to large numbers of classes, and learning descriptions of dynamically changing scenes. We have also initiated several new projects, such as dealing with learning shape descriptions, automated determination of "key" attributes, and developing autonomous vision agents.

Among the major topics to be investigated in the future is the development of a learning methodology capable of self-improving its knowledge representation space and automatically generating higher-level problem-relevant attributes (constructive induction). Other topics involve representing and learning imprecisely defined visual concepts, and demonstrating the usefulness of the proposed methods for a variety of problems of practical utility.

Acknowledgments

The authors thank Azriel Rosenfeld for helpful comments and discussions. This report includes research results by the authors and their collaborators Ali Hadjarian and Mark Maloof, who are Graduate Research Assistants at the GMU Center for Machine Learning and Inference.

References

Bala, J. and Michalski R., "Recognition of Textural Concepts Through Multilevel Symbolic Transformations," *Proceedings of the 3rd International Conference on Tools for Artificial Intelligence*, San Jose CA., Nov. 1991.

Bala, J., Michalski R.S., and Wnek J., "The Principal Axes Method for Constructive Induction," *Proceedings of the 9th International Conference on Machine Learning*, Aberdeen, Scotland, July 1992.

Bala, J. and Pachowicz P., "Recognizing Noisy Patterns Via Iterative Optimization and Matching of Their Rule Description," International Journal of Pattern Recognition and Artificial Intelligence , vol. 9, pp. 513-538. 1992.

Bala, J., Michalski, R.S. and Wnek, J., "The PRAX Approach to Learning a Large Number of Texture Concepts," *Technical Report FS-93-04, Machine Learning in Computer Vision: What, Why and How? , AAAI Fall Symposium on Machine Learning in Computer Vision*, AAAI Press, Menlo Park, CA, October 1993.

Bala, J. and Pachowicz, P.W., "Issues on Noise Tolerant Learning from Sensory Data," *Technical Report FS-93-04, Machine Learning in Computer Vision: What, Why and How?, AAAI Fall Symposium on Machine Learning in Computer Vision*, AAAI Press, Menlo Park, CA, October 1993.

Bloedorn, E., Wnek, J., and Michalski, R. S. "Multistrategy Constructive Induction: AQ17-MCI," *Reports of the Machine Learning and Inference Laboratory* MLI-93-4. Center for Artificial Intelligence, George Mason University, 1993.

Rivlin, E. and Rosenfeld, A., "Learning to Navigate in a Maze," to appear.

Herve, J.Y., "Learning Hand/eye Coordination by an Active Observer," to appear.

Kerber, R., "ChiMerge: Discretization of Numeric Attributes," *Proceedings AAAI-92*, 123–128, 1992.

Maloof, M. and Michalski R.S., "Learning Descriptions of 2D Shapes for Object Recognition and X Ray Images," *Reports of Machine Learning and Inference Laboratory, MLI-94-4*. Center for Artificial Intelligence, George Mason University, 1994.

Michalski, R. S., "A Variable-Valued Logic System as Applied to Picture Description and Recognition," in Graphic Languages, Nake, F. and Rosenfeld (eds.), A., North Holland, 1972.

Mozetic, I., "NEWGEM: Program for Learning from Examples, Program Documentation and User's Guide", *Reports of Machine Learning and Inference Laboratory, MLI-85-17*. Center for Artificial Intelligence, George Mason University, 1985.

Michalski, R.S, Mozetic, I., Hong, J., and Lavrac, N., "The Multipurpose Incremental Learning System AQ15 and its Testing Application to Three Medical Domains," *Proceedings of the 5th National Conference on Artificial Intelligence*, 1986.

Michalski, R.S., Bala, J.W. and Pachowicz, P.W., "GMU Research on Learning in Vision: Initial Results," *Proceedings of the ARPA Image Understanding Workshop*, Washington D.C., April, 1993.

Michalski, R.S., Rosenfeld, A., Pachowicz P. and Alomoinos, Y., "Preliminary Report, on the NSF/ARPA Workshop on Learning and Vision," *Proceedings of the ARPA Image*

Understanding Workshop, Washington D.C., April, 1993.

Pachowicz, P.W., "Local Characteristics of Binary Images and Their Application to the Automatic Control of Low-Level Robot Vision", *Computer Vision and Image Processing*, Special Issue, L. Shapiro and A. Rosenfeld (eds.), pp.67-102, Academic Press, 1992.

Pachowicz, P. W., "Invariant Object Recognition: A Model Evolution Approach," *Proceedings ARPA Image Understanding Workshop*, Washington D.C., pp.715-724, April 1993a.

Pachowicz, P. W., "Integration of Machine Learning and Vision into an Active Agent Paradigm", *Technical Report FS-93-04, Machine Learning in Computer Vision: What, Why and How?, AAAI Fall Symposium on Machine Learning in Computer Vision*, AAAI Press, Menlo Park, CA, October 1993b.

Pachowicz, P. W., "Semi-Autonomous Evolution of Object Models for Adaptive Object Recognition", *IEEE Trans. on Systems, Man and Cybernetics,* Vol. 24, 1994a (in press).

Pachowicz, P.W. and Bala, J., "A Noise-tolerant Approach to Symbolic Learning from Sensory Data," *Journal of Intelligent and Fuzzy Systems,* 1994b (in press).

Quinlan, J. Ross, "C4.5: Programs for Machine Learning", Morgan Kaufmann, San Mateo, CA, 1993.

Sydow, P. D., and Cooper, E. G., "Development of a Machine Vision System for Automated Structural Assembly," *NASA Technical Memorandum* 4366. Langley Research Center, Hampton, VA, 1992.

Weiss, S. M., and Kulikowski, C. A., "Computer Systems that Learn: Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning and Expert Systems," *Morgan Kaufmann*, San Mateo, CA, 1992.

Zurada, J. M., "Introduction to artificial neural systems," *West Publishing,* St. Paul, MN, 1992.